`Pines:` Bayes factors for selecting regression models

General Formulation

carlin:chib:95 consider the general problem of having *K* models with parameters $\theta_1, \ldots, \theta_K$, and wanting to obtain the posterior probability of each model. If the model indicator *M* is specified as a variable and hence as a node in the graph, *M* can then be sampled in a Gibbs run, and hence $\hat{p}(M = j\,|y)$ is obtained as a frequency of *M=j* in the sample. However, we need to specify a full probability model in order to satisfy MCMC conditions for convergence.

Their approach is to make the following assumptions:

- *y* is independent of $\theta_{k \neq j}$ given that *M=j*; *i.e.* *M* picks which parameters are relevant to *y*.
- $\theta_1, \ldots, \theta_K$ are independent given the model indicator *M*.

These imply an overall joint distribution

$$\begin{aligned} p(y, \underline{\theta}, M = j) &= p(y|\underline{\theta}, M = j)\, p(\underline{\theta}|M = j)\, p(M = j) \\ &= p(y|\theta_j, M = j) \times \prod_k p(\theta_k|M = j)\, p(M = j) \end{aligned}$$

When it comes to Gibbs sampling, the full conditional distributions are

$$\begin{aligned} p(M = j|\underline{\theta}, y) &\propto p(y, \underline{\theta}, M = j) \\ &= p(y|\theta_j, M = j) \times \\ &\quad \prod_k p(\theta_k|M = j)\, p(M = j) \end{aligned}$$

$$\begin{aligned} p(\theta_j|\theta_{\neq j}, y, M = j) &\propto p(y|\theta_j, M = j)\, p(\theta_j|M = j) \\ p(\theta_j|\theta_{\neq j}, y, M = k) &\propto p(\theta_j|M \neq j) \end{aligned}$$

$p(\theta_{k=j}|M \neq j)$ are known as *pseudo-priors*, and although their form is theoretically arbitrary, it is convenient to have them close to $p(\theta_j|M = j, y)$ so that plausible values are generated even when the model is being assumed false.

Carlin and Chib recommend a two-stage approach to estimation and model choice:

- Run each model separately using `estimation priors'.

- Use an approximation of the resulting posterior distributions as pseudo-priors for other models.
- Run sampler for all models together, monitoring *M*.
- Adjust the prior for *M* to ensure frequent visitation to all models.
- Re-adjust estimate of $p(M|y)$ to allow for the choice of prior on the model.

One of the examples of carlin:chib:95 concerns data of williams:59 on 42 specimens of radiata pine. For each specimen the maximum compressive strength y_i was measured, with its density x_i and its density adjusted for resin content z_i . Part of the data is shown below.

Specimen	strength y_i	density x_i	adjusted z_i
1	3040	29.2	25.4
2	2470	24.7	22.2
3	3610	32.3	32.2
4	3480	31.3	31.0
....			
41	3030	33.2	29.4
42	3030	28.2	28.2

Two alternative models are being considered:

$$\begin{array}{lcl} \text{Model 1: } y_i & \sim & \text{Normal}(\alpha + \beta x_i, \tau_1) \\ \text{Model 2: } y_i & \sim & \text{Normal}(\gamma + \delta z_i, \tau_2) \end{array}$$

The graph for the joint model is shown in Figure 20.

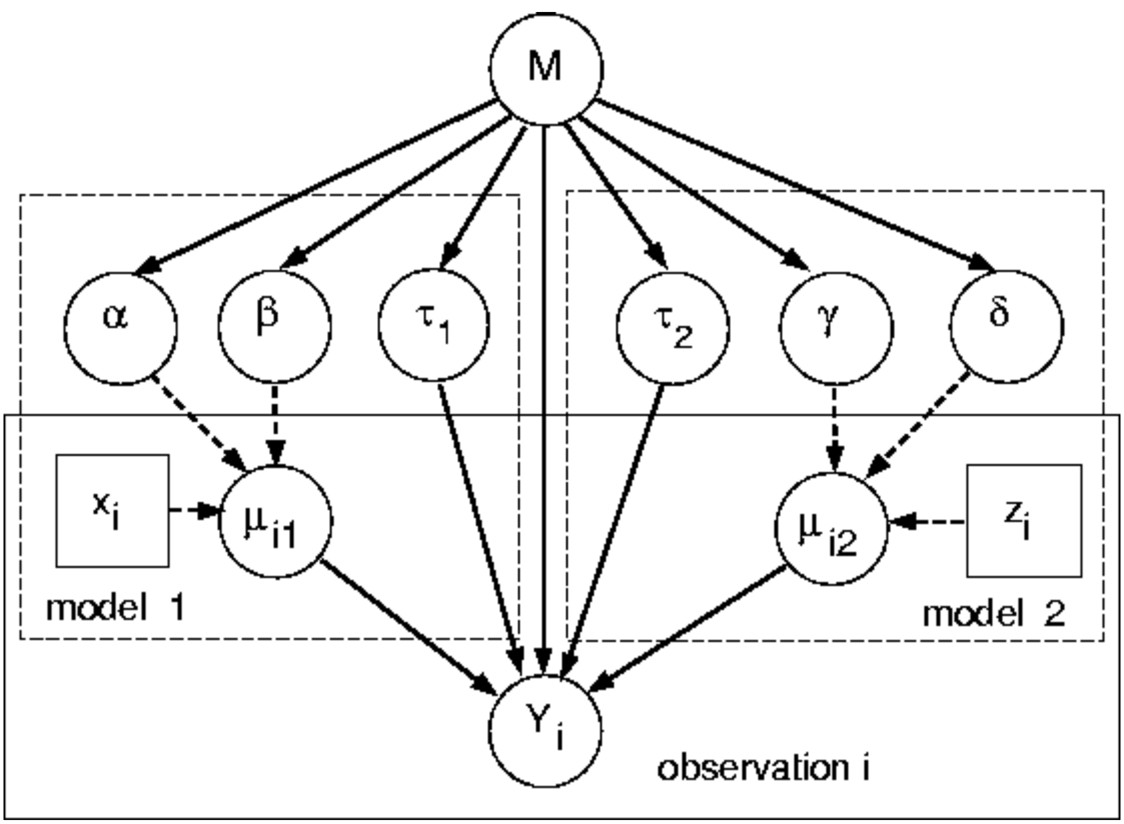

Figure 20: Graphical model for `pines` example showing the two models being simultaneously handled within a unified framework.

The following BUGS code shows that all variables were standardised to have mean 0 and variance 1 before analysis.

`pines` : model specification in BUGS

```
model pines;

const
   N = 42,  # number of data points
   M = 2;   # number of models
var
   Y[N],  Ys[N], # raw and standardised data
   x[N],  xs[N],
   z[N],  zs[N],
   mu[M,N],      # means for each model
   tau[M],       # precisions for each model
 alpha, mu.alpha[M], tau.alpha[M],  # priors for parameters
  beta, mu.beta[M] , tau.beta[M] ,
 gamma, mu.gamma[M], tau.gamma[M],
 delta, mu.delta[M], tau.delta[M],
 p[M],                               # prior for model
 pM2,                                # probability of model 2
  j,                                 # true model
   r1[M], l1[M],                     # priors for tau[1]
   r2[M], l2[M];                     # priors for tau[2]

data in "pines.dat";
inits in "pines.in";

{
# standardise data
   for(i in 1:N){
       Ys[i] <- (Y[i] - mean(Y[]))/sd(Y[]);
       xs[i] <- (x[i] - mean(x[]))/sd(x[]);
       zs[i] <- (z[i] - mean(z[]))/sd(z[]);
    }

# model node
       j  ~ dcat(p[]);
    p[1] <- 0.9995; p[2] <- 0.0005; # use for joint modelling
#   p[1] <- 1; p[2] <- 0 ; # include for estimating Model 1
#   p[1] <- 0 ; p[2] <-1;  # include for estimating Model 2
     pM2 <- step(j - 1.5);

# model structure
   for(i in 1:N){
       mu[1,i] <- alpha + beta *xs[i];
       mu[2,i] <- gamma + delta*zs[i];
       Ys[i]    ~ dnorm(mu[j,i],tau[j]);
   }

# Model 1
   alpha  ~ dnorm(mu.alpha[j],tau.alpha[j]);
   beta   ~ dnorm(mu.beta[j],tau.beta[j]);
   tau[1] ~ dgamma(r1[j],l1[j]);
# estimation priors
   mu.alpha[1]<- 0; tau.alpha[1] <- 1.0E-6;
   mu.beta[1] <- 0; tau.beta[1]  <- 1.0E-4;
   r1[1]      <- 0.0001;   l1[1] <- 0.0001;
# pseudo-priors
   mu.gamma[1] <- 0;  tau.gamma[1] <- 400;
   mu.delta[1] <- 1;  tau.delta[1] <- 400;
```

```
      r2[1]        <- 46     ;   l2[1] <- 4.5;

   # Model 2
      gamma   ~ dnorm(mu.gamma[j],tau.gamma[j]);
      delta   ~ dnorm(mu.delta[j],tau.delta[j]);
      tau[2] ~ dgamma(r2[j],l2[j]);
   # estimation priors
      mu.gamma[2] <- 0; tau.gamma[2] <- 1.0E-6;
      mu.delta[2] <- 0; tau.delta[2] <- 1.0E-4;
      r2[2]        <- 0.0001;   l2[2] <- 0.0001
   # pseudo-priors
      mu.alpha[2]<- 0; tau.alpha[2] <- 256;
      mu.beta[2] <- 1; tau.beta[2]  <- 256;
      r1[2]       <- 30    ;   l1[2] <- 4.5;
   }
```

Running each of the models separately gave the following within-model parameter estimates (posterior means and standard deviations).

	Model 1 (x)	Model 2 (z)
intercept	-.0001 $\pm$ .06	-.0002$\pm$ .05
gradient	.93 $\pm$ .06	.95 $\pm$ .05
$\tau = \sigma^{-2}$	6.8 $\pm$ 1.5	10.2$\pm$ 2.2

Approximations to these results are then used as the pseudo-priors for the `wrong' model shown in the BUGS code above: for Model 1 we set priors $\gamma \sim \text{Norm}(0,400)$, $\delta \sim \text{Norm}(1,400)$, $\tau \sim \text{Gamma}(46,4.5)$, while under Model 2 we set priors $\alpha \sim \text{Norm}(0,256)$, $\beta \sim \text{Norm}(1,256)$, $\tau \sim \text{Gamma}(30,4.5)$. The prior on the second model has to be adjusted to $p(M{=}2)$ = .0005 to ensure $M{=}1$ is visited frequently.

A BUGS run of 500 burn-in and 10000 iterations took 1 minute and gave $\hat{p}(M=2|y) = .629$. Hence the Bayes factor is $\frac{.629}{1-.629} \times \frac{.9995}{.0005} = 3389$, compared with Carlin and Chib's estimate of $\hat{p}(M=2|y) = .689$ and their Bayes factor of 4420. The differences in these results could be due to the different estimation priors used in our analysis.

---

next up previous contents

Next: Alli: multinomial-logistic models **Up:** BUGS 0.5 Examples Volume **Previous:** Beetles: logisticprobit and

Daniel Farewell
Mon Sep 13 16:39:37 BST 1999